REMARKS/ARGUMENTS

These remarks are made in response to the Office Action of November 23, 2003 (Office Action). As this response is timely filed within the 3-month shortened statutory period, no fee is believed due.

In paragraphs 1-25 of the Office Action, claims 1-23 have been rejected under 35 U.S.C. § 103(a) as being obvious over U.S. Patent Number 6,526,382 to Yuschik *et al.* (Yuschik) in view of the VoiceXML specification published 17/8/1999 (VoiceXML).

In response, Applicants have amended claims 1 and 12 to clarify that tapered prompts are to be provided in response to a previously determined quantity of correct responses. Applicants have also amended claims 2 and 13 to clarify that subsequent and different input requests are tapered request, as per FIGS. 1, 2, and 3 of the submission. Applicants have amended claims 5 and 17 to make them dependent upon claims 4 and 16, respectively. Applicants have amended claims 6 and 17, to make them dependant upon claims 1 and 12, respectively. Applicants have amended claims 9 and 21 to clarify that segments apply to application subject matter areas. Finally, Applicants have added new claims 24 and 25 to detail that the tailored prompts are delivered to subsequent and different input requests made within a particular application segment, as per the page 11, third paragraph of the submission. No new matter has been added as a result of these amendments.

Prior to addressing the rejections on the art, a brief review of the Applicants' invention is in order. The Applicants' claimed and disclosed subject matter teaches a system and a method for a speech-based interface to provide novice users with more robust speech prompting than expert users, wherein the shortened prompts provided to expert users are referred to as tapered prompts. This is accomplished by determining a quantity of correct and incorrect responses to delivered prompts. When a predetermined number of correct responses are received, a user can be considered an expert user and can be provided with tapered as opposed to full prompts. When an expert user provides an incorrect response to a tapered prompt, that user can be temporarily demoted to a novice user and can be subsequently provided with full prompts.

7

A speech interface can be segmented into different topical areas, where a user can be considered an expert within some topical areas and a novice within other topical areas. The user can receive either full or tapered prompts depending on the topical area for which the prompt is provided.

Turning to the rejections on the art, in paragraphs 1-25 of the Office Action, claims 1-23 have been rejected under 35 U.S.C. § 103(a) as being obvious over Yuschik in view of VoiceXML. Applicants respectfully disagree.

Yuschik discloses a method and system for a voice interface that decouples semantics from syntax. Voice recognition features can therefore be independently optimized for semantics and syntax. It is asserted that such a decoupling will improve recognition accuracy within an architecture designed to be extensible as to supported languages and cultural idiosyncrasies.

Yuschik is silent as to providing tapered prompts. Further, Yuschik does not teach that correct responses to prompts should be counted. Accordingly, Yuschik fails to teach that different programmatic actions should be taken whenever a predetermined number of correct responses have been received.

VoiceXML discloses a markup-based programming language that supports voice based interactions. VoiceXML, like any programming language, includes a multitude of standard programming constructs. These constructs include control statements (like While, For, If, Goto), data structures (like Boolean, Integer, and Counter data types), and exception handling features. Applicants agree that it is possible to write software code using VoiceXML (or any other programming language) that accomplishes any of a multitude of tasks. This capability, however, does not render software written in VoiceXML obvious under 35 U.S.C. § 103(a), in absence of specific teachings to construct the software in a particular fashion.

Referring to paragraph 3 of the Office Action, Claim 1 discloses:

> A categorized speech-based interface, comprising a prompt delivery system for delivering user prompts to users, each user prompts comprising at least one of a full prompt and a tapered prompt, said interface further comprising logic for determining the quantity of at least one of correct and incorrect responses to said user prompts, said prompt delivery system delivering

8

tapered prompts in response to the determination of a predetermined quantity
of said correct responses.

Applicants do not dispute that Yuschik discloses a voice activated user interface.
Yuschik, as acknowledged in the Office Action, is silent in regards to tapered prompts. Further,
Yuschik is silent as to counting correct responses to prompts. VoiceXML fails to cure the
deficiencies of Yuschik.

VoiceXML discloses an operational feature of VoiceXML of "tapered prompts" defined
as prompts that change with use, where "use" refers to prompts with an associated counter that
increases each time a prompt is repeated within a menu or field block. That is, the counter is
incremented automatically for each loop occurring within a menu or field block. The occurrence
of such a loop is an error avoidance measure implemented whenever an input is not understood
due to the occurrence of either a "noinput" or "nomatch" event, where "noinput" and/or
"nomatch" events are system level VoiceXML events that can occur responsive to user input.
The prompt count is, therefore, an error occurrence count inherent in the prompting structure of
the VoiceXML language.

VoiceXML does state that information-requesting prompts may become terser under the
assumption that the user is being more familiar with the task. Applicants assume that the
reference for "being more familiar with the task" (made in reference to an prompt counter that
increments upon an error condition) refers to an assumption that after receiving a multitude of
identical prompts each time an input was erroneously entered, a terser prompt can be provided.
VoiceXML does not teach that terser prompts should be provided responsive to correct
responses, nor does VoiceXML teach or suggest that correct responses to prompts should be
tracked in any fashion. Moreover, VoiceXML fails to teach or suggest that terser prompts are to
be provided in a forward-looking fashion, meaning all subsequent prompts for new questions
should be terse prompts as opposed to full prompts. Instead, VoiceXML only teaches that a terse
prompt can be provided responsive to the iterative receipt of incorrect responses to a single
question. Applicants emphasize that the data structure from which the teachings of prompt
counters is provided in VoiceXML is inherently limited to counting incorrect responses (not

9

correct ones) and for providing divergent prompts for a single question (not providing terse prompts in a forward-looking fashion to new questions posed subsequent to a predetermined quantity of correct responses).

VoiceXML does not teach or suggest that correct or incorrect responses should be recorded for application-level tasks. One of ordinary skill in the art of software programming is aware of modular coding practices and of variable persistence. Temporary variables of local scope should not be used to effect global application changes. The prompt counter is an internal counter that is reset each time the field or menu block is entered. Switching between multiple forms as suggested by the examiner in paragraph 3, would cause the prompt counter to reset for each switch. Thus, the application could not track the value of the prompt counter between application-level interactions using the local prompt counter. Any attempt to do so would violate basic programming principles and would not be done by one of ordinary skill in the art. Further, non-modular use of the Goto control mechanism violates accepted software design principles. Such potential violations are one reason that many modern, strongly typed programming languages do not include the flexible, but often misused Goto control mechanism.

Accordingly, one of ordinary skill in the art would not apply the prompt counter taught by VoiceXML to handle application-level concerns, such as globally altering a level of prompting provided to a user.

To reiterate, VoiceXML fails to teach or suggest that a quantity of correct or incorrect responses should be tracked for application-level functions. Moreover, VoiceXML fails to teach or suggest that correct responses to prompts should be tracked. Finally, VoiceXML fails to teach or suggest tapered prompts in the context used by the Applicants. Consequently, VoiceXML fails to cure the deficiencies of Yuschik.

Claim 12 has been rejected for the same grounds as claim 1 was rejected. Accordingly, Applicants assert that claim 12 should be allowable for the reasons provided above.

Regarding claim 2 and 13, delivering tapered prompts for subsequent and different input request made within an application is an application-level function. Yucshik is silent as to

10

tapered prompts, and as emphasized above, the prompt counter taught by VoiceXML is a local, as opposed to an application-level procedure.

Referring to paragraph 11, claim 9 discloses:

> The categorized speech-based interface of claim 1, wherein said interface is segmented, and said prompt delivery system delivers tapered prompts in a segment in response to a determination that correct responses have been provided by a user to a prerequisite quantity of said prompts while in said segment.

In the Office Action, Yuschik is cited for the proposition of using segmented interfaces. Column 18, lines 33-36 of Yuschik states:

> The grammar template 700 then breaks the set of menu selections into connectional "chunks" of between 2 and 4 choices presented as a group.

Yuschik teaches that larger menus can be partitioned into smaller groups of choices so as to not overload a users auditory short term memory. In contrast, Applicants teach that the counting of correct responses determining whether a user is proficient with an application. Proficiency can take into account application interface segmentations.

That is, if a user who has successful experiences within one part of an application (and is therefore receiving tapered prompts within that application part) moves to a new major application segment in which there has been no successful or unsuccessful interactions, the interface should start a new set of measurements for the new application segment. Accordingly, the user can be provided full prompts when operating within the new segment until that user provides a predetermined number of correct responses within the new segment; at which time, the user can be provided tapered prompts while operating within the new segment.

Claim 21 has been rejected for the same grounds as claim 9 was rejected. Accordingly, Applicants assert that claim 21 should be allowable for the above reasons.

In light of the above, withdraw of the 35 U.S.C. § 103(a) rejection with regard to claims 1-25 is respectfully requested. Applicants believe that this application is now in full condition for allowance, which action is respectfully requested. Applicants request that the Examiner call

11

{WP168330;1}

the undersigned if clarification is needed on any matter within this Amendment, or if the Examiner believes a telephone interview would expedite the prosecution of the subject application to completion.

Respectfully submitted,

Date: 17 Feb 2004

Gregory A. Nelson, Registration No. 30,577
Kevin T. Cuenot, Registration No. 46,283
Brian K. Buchheit, Registration No. 52,667
AKERMAN SENTERFITT
Post Office Box 3188
West Palm Beach, FL 33402-3188
Telephone: (561) 653-5000

12

{WP168330,1}